WHAT IS CLAIMED IS:

1. A system of virtual router domains comprising:

a host router running a common operating system;

a plurality of virtual router domains and processes logically partitioned within said host router, each said virtual router domain having a unique domain ID address and an independent replica array of all virtualized variables across said common operating system, each said process running in a said virtual router domain independently of all other said virtual router domains on top of said common operating system; and

said global variables being accessed by macro references in each said virtual router domain.

2. The system of claim 1 wherein said common operating system runs on a master control processor within said host router.

3. The system of claim 2 wherein said common operating system is a version of FreeBSD.

4. The system of claim 1 wherein said common operating system manages the reporting of hardware failures across all virtual router domains of said host router.

5. The system of claim 1 wherein said plurality of processes comprise routing software applications.

6. The system of claim 5 wherein said plurality of processes comprise independent plural identical copies of at least one said process.

7. The system of claim 5 wherein said plurality of processes comprise a copy of a dynamic routing protocol (DRP) software application.

8. The system of claim 1 wherein said plurality of processes comprise a copy of a SNMP application.

9. The system of claim 1 wherein said macros generate an array of said global variables when said virtual router domain is configured in.

10. The system of claim 9 wherein said macros generate scalar global variables when said virtual router domain is deconfigured.

11. The system of claim 1 further comprising a plurality of interfaces partitioned interchangeably among said virtual router domains, such that a particular interface is associated with only one such virtual router domain at one time, but can be repartitioned among said virtual router domains to reconfigure said host router.

12. The system of claim 11 wherein during said reconfiguring network traffic is removed from said interfaces that are repartitioned.

13. The system of claim 11 wherein said interface contains the unique domain ID address of said virtual router domain with which said interface is associated.

14. The system of claim 13 wherein said interface is an interface port of said host router.

15. The system of claim 14 wherein said host router comprises at least 320 said interface ports.

16. The system of claim 14 further comprising a socket created by at least one said process, said socket being associated exclusively with the virtual router domain in which it is created and containing said unique domain ID address of said domain in which it is created.

17. The system of claim 16 wherein multiple sockets are created by said at least one process in at least one said virtual router domain, such that each of said multiple sockets is associated exclusively with the virtual router domain in which said socket is created.

18. The system of claim 17 wherein said at least one process is movable from one said virtual router domain to a different said virtual router domain, such that said at least one process creates a said socket in each of at least two said virtual router domains.

19. The system of claim 17 wherein a particular socket associated with a particular virtual router domain is applied exclusively to live traffic networking independently of any other said virtual router domain of said host router.

20. The system of claim 17 wherein a particular socket associated with a particular virtual router domain is applied exclusively to a test bed operation independently of any other said virtual router domain of said host router.

21. The system of claim 16 wherein each of said virtual router domains maintains an independent routing table.

22. The system of claim 21 wherein each said socket uses the routing table of said virtual router domain in which said socket is created.

23. The system of claim 21 wherein said two distinct virtual router domains use the same Internet Protocol (IP) addresses without conflicting.

24. The system of claim 21 wherein one particular virtual router domain within said host router contains routing tables exclusively for internal interface addresses within said host router independently of any other said virtual router domain of said host router.

25. The system of claim 21 wherein a particular virtual router domain within said host router contains routing tables exclusively for interfaces externally visible from outside said host router independently of any other said virtual router domain of said host router.

26. The system of claim 21 wherein a failure of one of said plurality of said virtual router domains does not adversely affect a different one of said plurality of said virtual router domains.

27. A method of logically partitioning a host router into virtual router domains, comprising

configuring the kernel of a single common operating system running in said host router;

configuring in a plurality of virtual router domains within said host router;

identifying each said virtual router domain by a unique domain index number;

generating an independent identical set of replica arrays of global variables for each virtual router domain; and

associating a process with each said virtual router domain of said host router, such that said processes run in said virtual router domains independently of one another on top of said single common operating system of said host router.

28. The method of claim 27 wherein said global variables are generated by macros.

29. The method of claim 28 wherein said macros generate arrays of global variables when said virtual router domain is configured in within said host router.

30. The method of claim 29 wherein said macros generate scalar global variables when said virtual router domain is deconfigured.

31. The method of claim 27 wherein said single common operating system runs on a master control processor within said host router.

32. The method of claim 31 wherein said single common operating system is a version of FreeBSD.

33. The method of claim 27 wherein said processes comprise routing software applications.

34. The method of claim 33 further comprising independently running plural identical copies of at least one said process.

35. The method of claim 33 wherein said processes comprise a copy of a dynamic routing protocol (DRP) software application.

36. The method of claim 27 wherein each said process manages an instantiation of a common networking code.

37. The method of claim 27 further comprising partitioning a plurality of host router interfaces interchangeably among said virtual router domains, such that a particular interface is associated with only one such virtual router domain at one time.

38. The method of claim 37 further comprising repartitioning said plurality of interfaces among said virtual router domains, such that said host router is reconfigured.

39. The method of claim 38 wherein during said reconfiguring network traffic is removed from said interfaces that are repartitioned.

40. The method of claim 37 wherein said interface contains the unique domain index number of said virtual router domain with which said interface is associated.

41. The method of claim 40 wherein said interface is an interface port of said host router.

42. The method of claim 41 wherein said host router comprises at least 320 said interface ports.

43. The method of claim 41 wherein said process creates a socket, such that said socket is associated permanently and exclusively with the virtual router domain in which it is created and containing said unique domain index number of said domain in which it is created.

44. The method of claim 43 wherein multiple sockets are created by at least one said process in at least one said virtual router domain, such that each of said multiple sockets is associated permanently and exclusively with the virtual router domain in which said socket is created.

45. The method of claim 44 further comprising moving said at least one process from one said virtual router domain to a different said virtual router domain, such that said at least one process creates a said socket in each of at least two said virtual router domains.

46. The method of claim 44 wherein said process maintains a file descriptor table containing pointers to said sockets associated with said virtual router domain.

47. The method of claim 46 wherein a particular socket associated with a particular virtual router domain is applied exclusively to live traffic networking independently of any other said virtual router domain of said host router.

48. The method of claim 46 wherein a particular socket associated with a particular virtual router domain is applied exclusively to a test bed operation independently of any other said virtual router domain of said host router.

49. The method of claim 43 wherein each of said virtual router domains maintains an independent routing table.

50. The method of claim 49 wherein each said socket uses the routing table of said virtual router domain in which said socket is created.

51. The method of claim 49 wherein said two distinct virtual router domains use the same Internet Protocol (IP) addresses without conflicting.

52. The method of claim 49 wherein one particular virtual router domain within said host router contains routing tables exclusively for internal interface addresses within said host router independently of any other said virtual router domain of said host router.

53. The method of claim 49 wherein a particular virtual router domain within said host router contains routing tables exclusively for interfaces externally visible from outside said host router independently of any other said virtual router domain of said host router.

54. The method of claim 49 wherein a failure of one of said plurality of said virtual router domains does not adversely affect a different one of said plurality of said virtual router domains.